

THIS IS A REFERENCE SYLLABUS DESCRIBING AN NKU COURSE IN GENERAL. ENROLLED STUDENTS SHOULD CONSULT THE ACTUAL SYLLABUS OF THE COURSE IN WHICH THEY ARE REGISTERED.

## CSC 464 Design and Analysis of Algorithms

### CATALOG DESCRIPTION:

**CSC 464 Design and Analysis of Algorithms (3,0,3)** Proofs of time and space bounds on important algorithms; advanced algorithms on graphs, sequences and sets, divide-and-conquer and dynamic programming; randomized algorithms; parallel algorithms. PREREQ: C or better in CSC 364 and MAT 385.

**LAST TAUGHT:** Fall 2008 (J. Ward)

**SCHEDULED LAB USAGE:** None

### STUDENT BACKGROUND EXPECTATIONS:

1. Knowledge of basic data structures such as linked lists, stacks, queues, search trees, and hash tables (from CSC 364: Data Structures and Algorithms).
2. Familiarity with recurrence relations, proof by induction, basic set theory, Boolean algebra, and graphs (from MAT 385: Discrete Mathematics).
3. Familiarity with series and limits (from MAT 120: Calculus I and MAT 220: Calculus II, which are prerequisites for MAT 385).

### CORE TOPICS COVERED:

- Big-O notation (as well as Big- $\Theta$  and Big- $\Omega$ )
- Divide-and-conquer algorithms
- Expressing runtime with recurrence relations, and deriving corresponding Big-O analyses
- Decompositions of graphs into connected and strongly connected components
- Finding shortest paths in graphs
- Finding minimum spanning trees
- Greedy algorithms
- NP-complete problems (polynomial time reductions and heuristics)
- Dynamic programming
- Set and sequence algorithms
- Randomized algorithms
- Parallel algorithms

### MOST RECENT TEXTBOOK USED :

*Algorithms*, Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani (Prentice Hall).  
Chapters Covered: 0-6, 8-9.

### SOFTWARE REQUIRED:

None, though it is assumed that students have access to a programming environment of their choice.

### STUDENT WORK

Homework assignments. Programming assignments. In-class exams.

### LEARNER OUTCOMES

Students will be able to...

1. Students will be able to describe and to trace the execution of classic computer algorithms.
2. Students will understand some of the strategies used in algorithm design such as divide-and-conquer, dynamic programming, and greediness.
3. Students will be able to analyze the asymptotic time and space complexity of various algorithms.
4. Students will gain experience in implementing efficient algorithms.

### CROSS-LISTING

CSC 564